

Simple Real-Time Strategy (SRTS) Pack

Made by Ray McClain

Table of Contents

1. Preface
2. Feature List
3. Input Setup
4. Camera Movement
5. UGrid
6. Resources
7. Unit Selection
8. Factions
9. Unit Creation
10. A* Movement
11. Fog Of War
12. MiniMap
13. Tech Tree
14. Buildings
15. Types
16. Cursor Manager
17. GUI Manager
18. Misc

Preface

Thank you for purchasing my SRTS pack. This pack should assist you in getting all the basics for your RTS game. There are quite a few features included in the pack, however there are certain features that I did decide against including, such as unit formations or networking implementation. I do plan on adding features in the future though, so it is possible that either of those features could be added in. Please stay tuned for any updates!

Feature List :

This script pack includes functional scripts for :

1. UGrid (Grid Generator System)
2. Simple Map Seeding
3. MultiThreaded A* Pathfinding
4. Camera Movement
5. Unit/Building Selection
6. Unit Controller
7. Factions
8. Unit Types
9. Building Controller
10. Building Placement
11. Tech Trees and Effects
12. Fog of War system
13. Mini Map System
14. Basic Cursor Manager
15. GUI Manager
16. Resource System

And Editors for the following scripts :

1. UGrid
2. Faction Editor

Input Setup

Your Unity input should include these inputs :

▼ Horizontal	▼ Vertical		
Name	Horizontal	Name	Vertical
Descriptive Name		Descriptive Name	
Descriptive Negative		Descriptive Negative	
Negative Button	left	Negative Button	down
Positive Button	right	Positive Button	up
Alt Negative Button	a	Alt Negative Button	s
Alt Positive Button	d	Alt Positive Button	w
Gravity	1000	Gravity	1000
Dead	0.001	Dead	0.001
Sensitivity	1000	Sensitivity	1000
Snap	<input type="checkbox"/>	Snap	<input type="checkbox"/>
Invert	<input type="checkbox"/>	Invert	<input type="checkbox"/>
Type	Key or Mouse Button	Type	Key or Mouse Button
Axis	X axis	Axis	X axis
Joy Num	Get Motion from all Joysticks	Joy Num	Get Motion from all Joysticks

▼ IncreaseMoveSpeed	▼ Mouse ScrollWheel		
Name	IncreaseMoveSpeed	Name	Mouse ScrollWheel
Descriptive Name		Descriptive Name	
Descriptive Negative		Descriptive Negative	
Negative Button		Negative Button	
Positive Button	x	Positive Button	
Alt Negative Button		Alt Negative Button	
Alt Positive Button		Alt Positive Button	
Gravity	1000	Gravity	1000
Dead	0.001	Dead	0.001
Sensitivity	1000	Sensitivity	5
Snap	<input type="checkbox"/>	Snap	<input type="checkbox"/>
Invert	<input type="checkbox"/>	Invert	<input type="checkbox"/>
Type	Key or Mouse Button	Type	Mouse Movement
Axis	X axis	Axis	3rd axis (Joysticks and ScrollWheel)
Joy Num	Get Motion from all Joysticks	Joy Num	Get Motion from all Joysticks

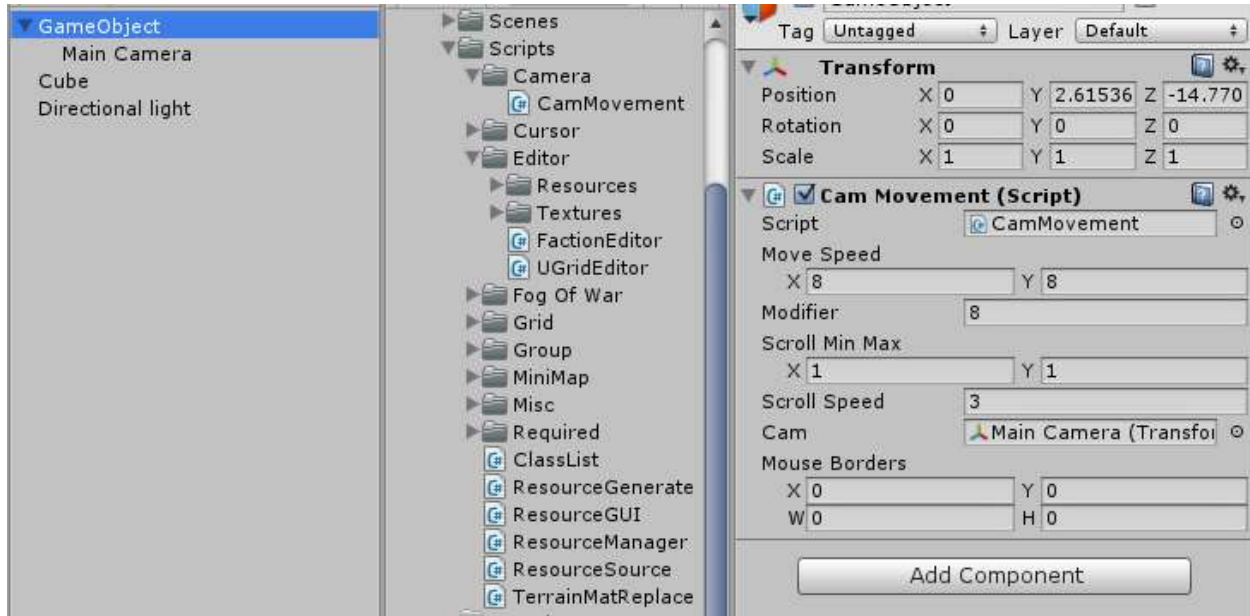
▼ LMB		▼ RMB	
Name	LMB	Name	RMB
Descriptive Name		Descriptive Name	
Descriptive Negative		Descriptive Negative	
Negative Button		Negative Button	
Positive Button	mouse 0	Positive Button	mouse 1
Alt Negative Button		Alt Negative Button	
Alt Positive Button		Alt Positive Button	
Gravity	3	Gravity	0
Dead	0.001	Dead	0.001
Sensitivity	3	Sensitivity	3
Snap	<input checked="" type="checkbox"/>	Snap	<input checked="" type="checkbox"/>
Invert	<input type="checkbox"/>	Invert	<input type="checkbox"/>
Type	Key or Mouse Button	Type	Key or Mouse Button
Axis	X axis	Axis	X axis
Joy Num	Get Motion from all Joysticks	Joy Num	Get Motion from all Joysticks

▼ ContinuePlace	
Name	ContinuePlace
Descriptive Name	
Descriptive Negative	
Negative Button	
Positive Button	left shift
Alt Negative Button	
Alt Positive Button	right shift
Gravity	1000
Dead	0.001
Sensitivity	1000
Snap	<input type="checkbox"/>
Invert	<input type="checkbox"/>
Type	Key or Mouse Button
Axis	X axis
Joy Num	Get Motion from all Joysticks

Camera Movement

The Camera Movement system is rather simple to set up, you will need a Camera GameObject, an Empty GameObject, and the CamMovement script found under Scripts/Camera/CamMovement.cs

First, move the Empty GameObject to the same location as the Camera GameObject on the XYZ axis. Rotate the Camera GameObject to the view you want in the game and make it a child of the Empty GameObject. Finally, add the CamMovement script to the Empty GameObject and modify the values accordingly.



Variables

Move Speed : The speed the camera can move on both axis

Modifier : When the IncreaseMoveSpeed input is held, this will be the multiplier to the MoveSpeed. If yours are as shown above, then the final speed the camera will be moving at is 24, or $8*3$.

Scroll Min Max : The in and out maximums that you can scroll. In the above example, you can scroll out 1 and in 1.

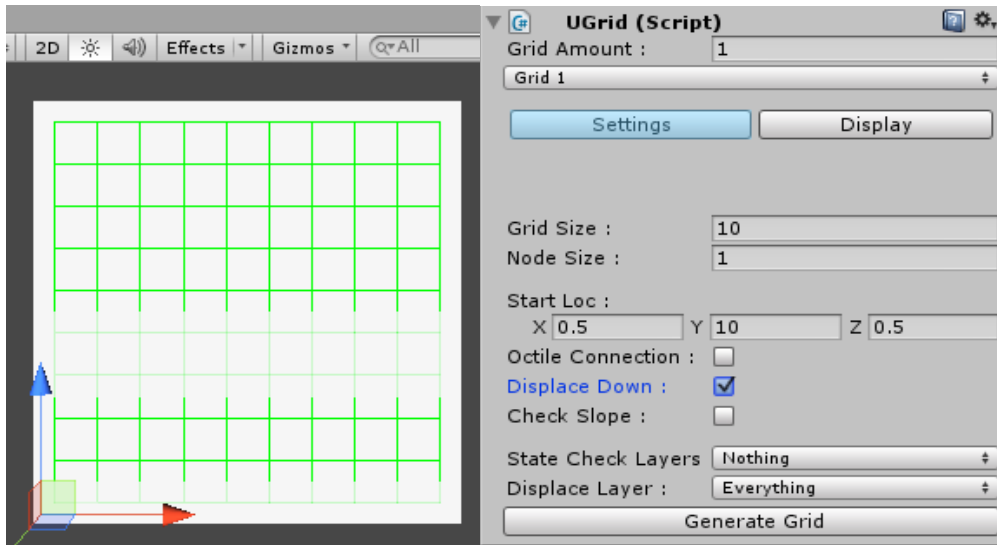
Scroll Speed : The speed that you are scrolling

Cam : The Main Camera GameObject that is the child of this GameObject

Mouse Borders : If you would like the mouse to move the Camera by moving to the borders, set these to the desired border locations

UGrid

The UGrid is the basis for RTS functionality. It will handle Seeding, A* Pathfinding, and Building Placement. For the UGrid you are going to need to make a new **Empty GameObject** and simply attach the UGrid component to it. The name of the object will change automatically to UGrid and will automatically revert if you attempt to change it. This is due to the grid being found through **GameObject** name. Following that, simply enter the grid information for your game, although the smallest you can make the grid without losing detail in your game the better for performance.



Variables

Grid Amount : The amount of grids you would like to have on the map.

Grid Size : The X and Y amount of gridpoints on the grid.

Node Size : The distance between gridpoints on the grid.

Start Loc : The location that this grid starts at.

Octile Connection : Instead of simply using the gridpoints to the left, right, above, and below this will add the above-right, above-left, below-right, and below-left. This is for A* purposes.

Displace Down : This will set the points Y location to whatever the first collision on the displace layer that is seen below. If you use this, then please set your Start Loc's Y to above your object.

Check Slope : A Boolean that will check the slope of the points before declaring them open or closed.

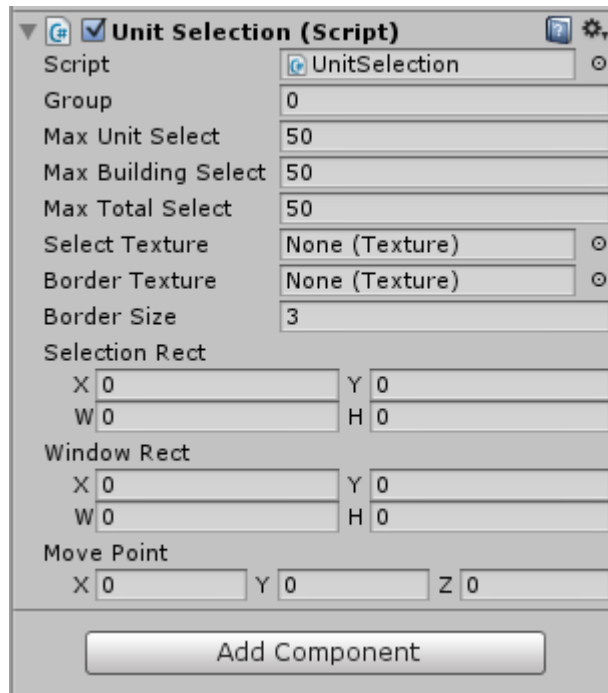
Slope Max : The maximum accepted slope.

State Check Layer : The layers that we check the collisions of to see if the points are open and closed

Displace Layer : The layers that we check the collisions from the downward raycast to see if we can move the points down on the Y-axis

Unit Selection

Create another **Empty GameObject** and apply the Unit Selection script to it.



Variables

Group : This will be automatically changed to the group that you are working with

Max Unit Select : The maximum amount of units you can select

Max Building Select : The maximum amount of buildings you can select

Max Total Select : The maximum amount of Buildings and Units you can select

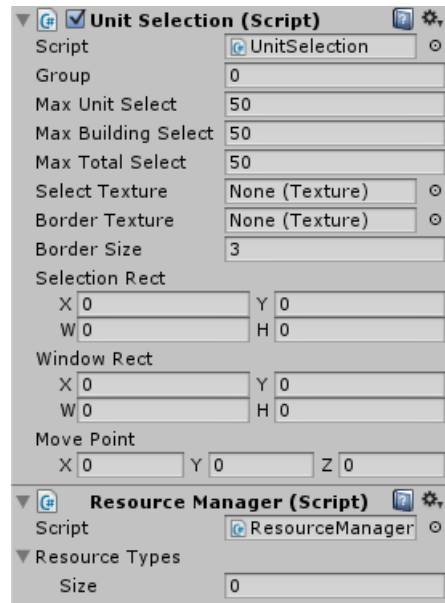
Select Texture : The texture you are using for the interior of the selection box. I use a 1 pixel transparent texture for simplicity.

Border Texture : The texture used as the border of the Selection Rectangle

Window Rect : The rectangle that you can select in. Used if you have a GUI at the bottom of your screen that you don't want to be able to select over.

Resource Manager

Add the Resource Manager script to the same GameObject as the Unit Selection script is attached, as shown :



Variables

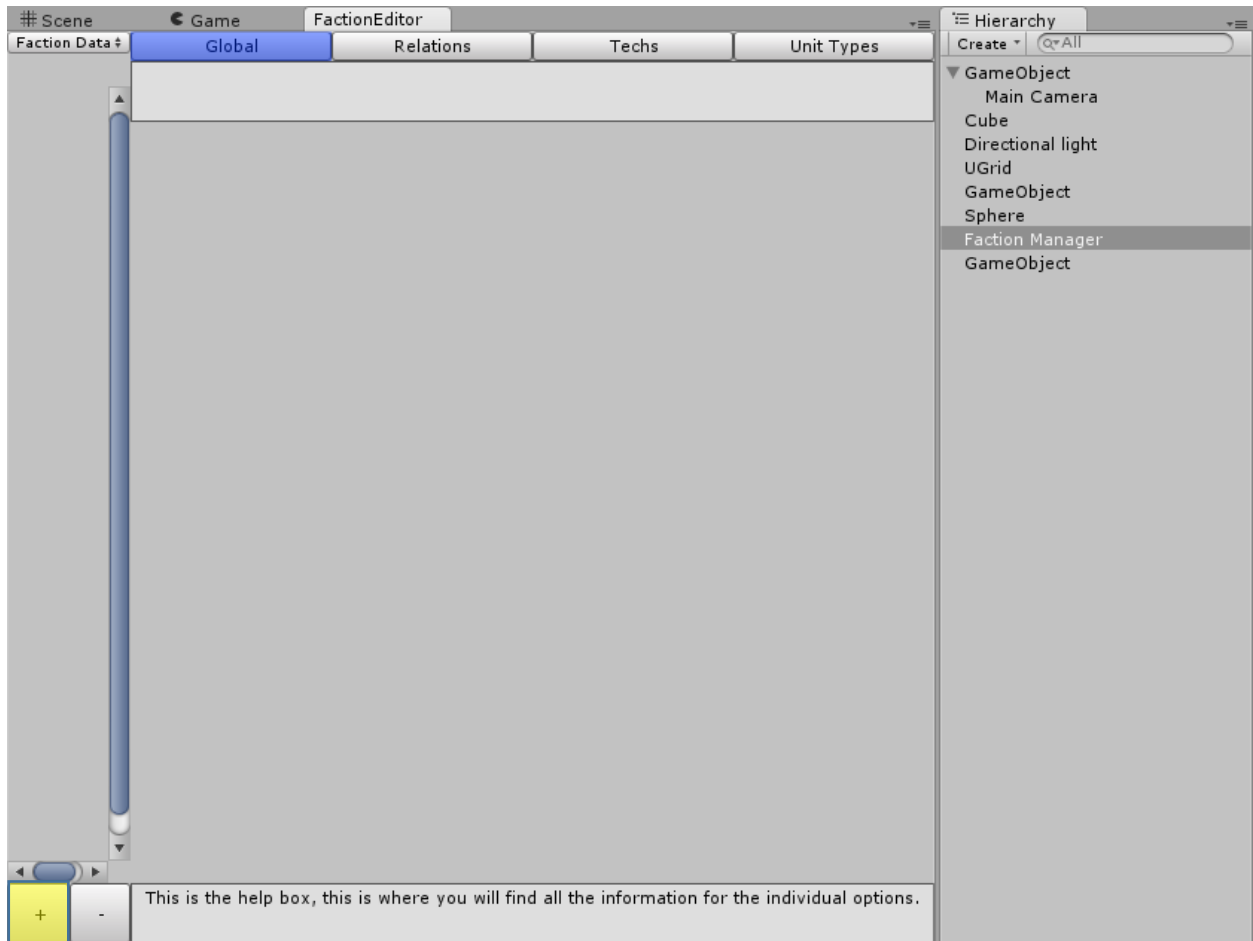
Resource Types : A custom class option with the variables Name and Amount

Name : The name of the resource to display

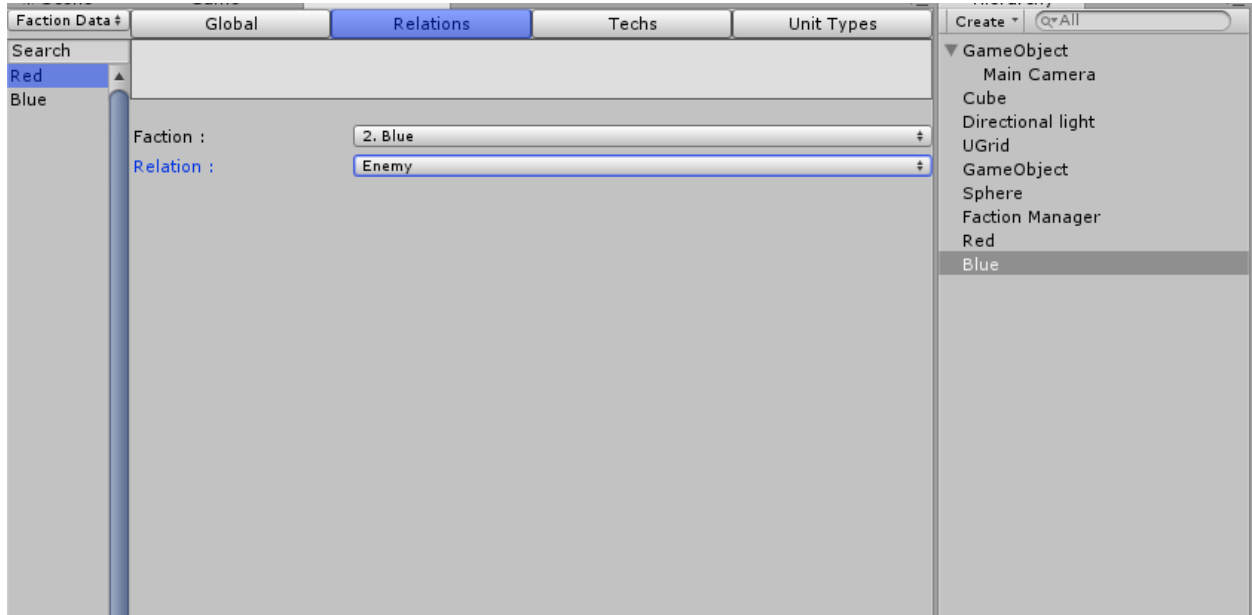
Amount : The amount of the resource the player currently has

Factions

So lets get started with the Faction system. Go ahead and make an **Empty GameObject** and apply the Group Manager script to it, found under Scripts/Group/GroupManager.cs. The name will change to Faction Manager, do not alter that. Next create another **Empty GameObject** and apply the Group script to it found in the same location. Now you can go to Window/Faction Manager to open the Faction Manager Editor. A window should open that looks like this :



I recommend putting it on a separate monitor or maximizing it as much as possible. Now you can add the GameObject with the Group script on it to the Faction Manager system by clicking the + Button highlighted above. That will create a field for an GameObject in the editor, which you will then drag your new faction GameObject to. Now that your Faction has been added, go ahead and name it in the editor. You can create another faction GameObject to set up their Relations to each other as depicted below.



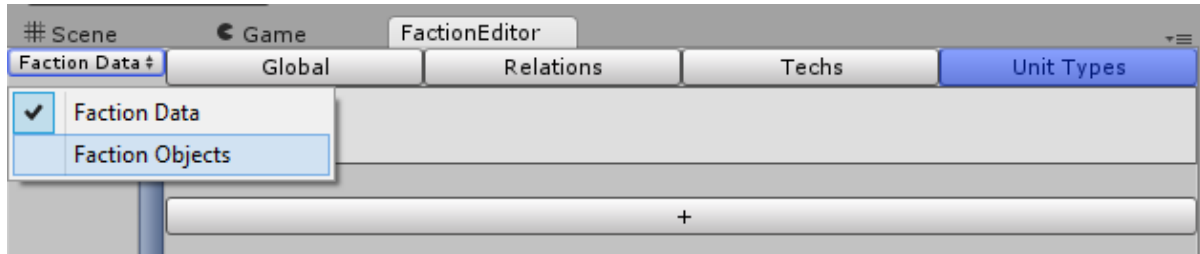
Variables

Name : The name of the Faction

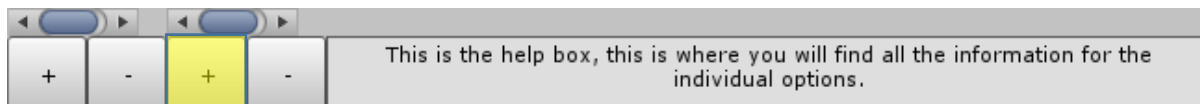
Relation : The relation variable has 3 options, Ally, Neutral, and Enemy. As far as I know the Neutral option currently doesn't do anything, so I recommend using either Ally or Enemy.

Unit Creation

Now it is time to make your first unit. Go ahead and click the Faction Data option in the top left corner of the Faction Editor window and select Faction Objects.



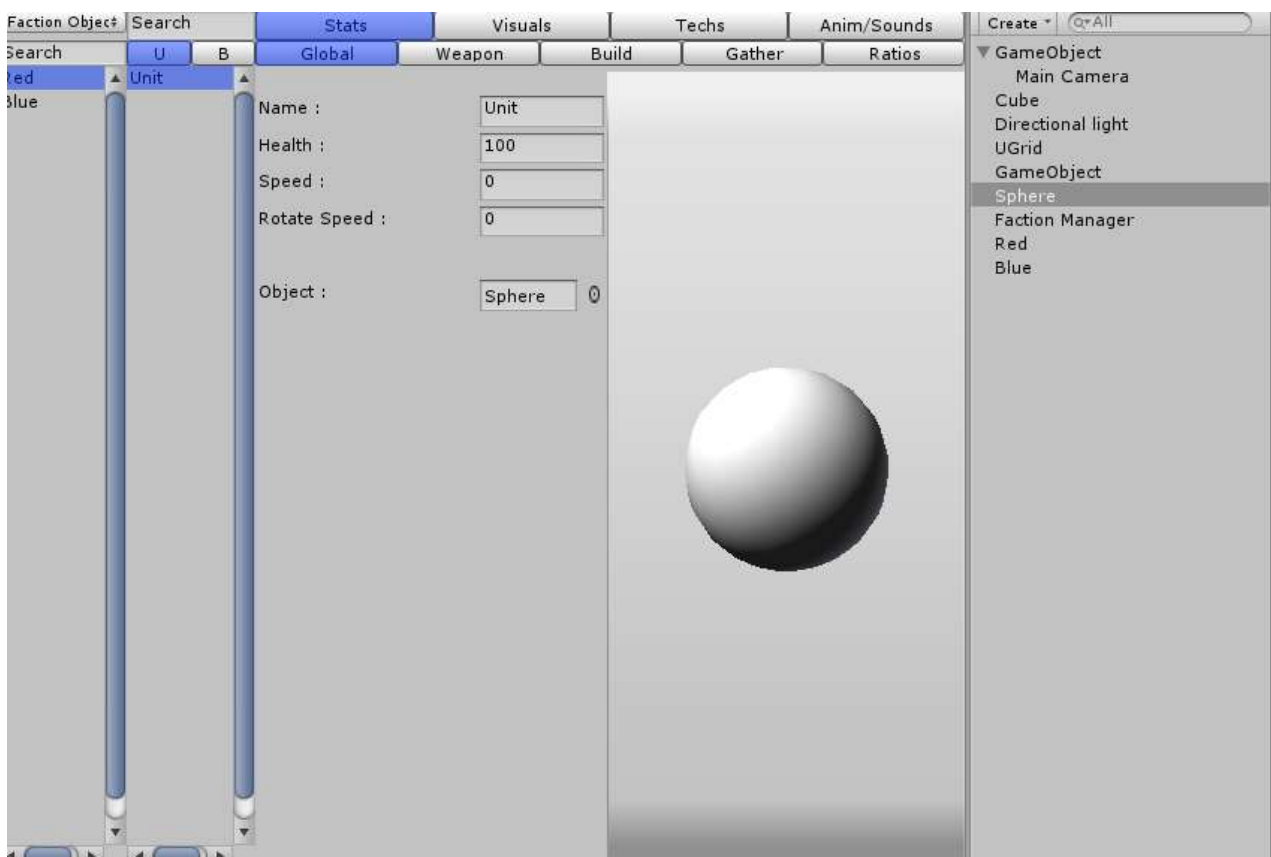
On the new screen you should select the Faction you want to modify and go on the bottom to the second set of + and - Buttons and add an Empty Unit.



Add your base model to the GameObject Field that appeared on the window and click the large + Button that will appear on the right (if it doesn't appear that means that your model already has the Unit Controller script applied).

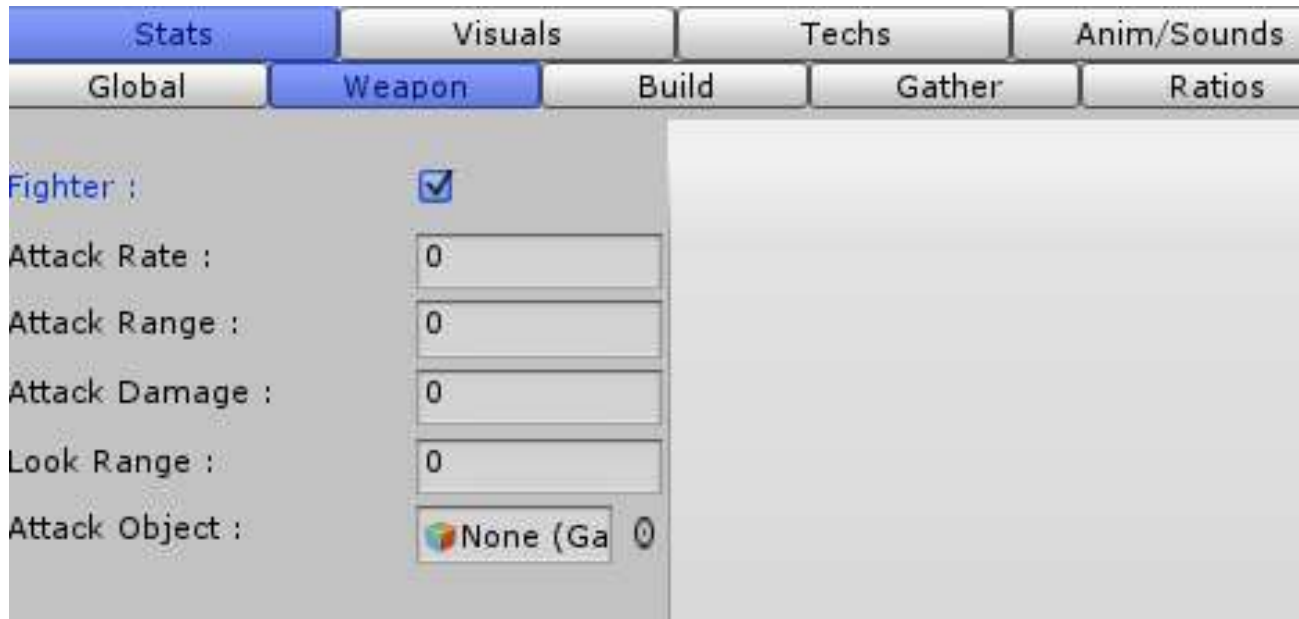
When you add the Unit Controller script it will also add a Unit Movement, Rigidbody, Animator, and Audio Source component automatically. These are all required for the Unit Controller to be fully functional.

The window should now look like this :



Unit Creation (cont.)

You can set the Unit as a Fighter, Builder, or Gatherer simply by going to the Stats menu and then selecting the corresponding option and clicking check. It will bring up a set of options similar to the example shown below :



You can also add the MiniMapSignal script and VisionSignal scripts through the Faction Editor by going to Visuals then MiniMap/Vision and clicking the Add option.

Variables

Under Global

Name : The Name of the Unit as it is displayed in-game

Max Health : The Maximum amount of Health the unit can have

Health : The starting health the unit has

Speed : The movement speed of the unit

Rotate Speed : The rotational speed of the unit

Type : If you have Unit Types set up, this will allow you to set the type of the unit, otherwise it won't appear as an option

Object : The Unit Object is replaceable here

Under Weapon

Fighter : Is this unit a fighter?

Attack Rate : The rate at which this unit can attack

Attack Range : The range the unit can attack within

Attack Damage : The damage the unit's attack will do (before Type comes into play)

Look Range : The range that the unit will see the enemy unit and start following them in

Attack Object : The object that is created when the unit attacks

Under Builder

Builder : Is this unit a builder?

Building (Popup) : The Building object that is being referred to. The Buildings option is reliant on the Buildings in the current faction

Can Build : Can the unit build this Building?

Build Amount : The amount the unit builds of this Building per rate.

Build Rate : The rate at which the unit builds the Building.

Under Gather

Gatherer : Is this unit a gatherer?

Resource (Popup) : The Resource that is being referred to. The Resource option is reliant on the Resource Manager's Resource Types.

Can Gather : Can the unit gather this Resource?

Gather Amount : The amount the unit gathers per rate.

Gather Rate : The rate at which the unit gathers this Resource.

Under Ratios

:

Under Selected

GUI Image : The Image that is displayed in the Current Unit Selection area for this unit.

Objects : The Objects that activate and deactivate depending on the Units selection state.

Under MiniMap

MiniMap Signal : Does this unit appear on the minimap?

Tag : The MiniMap tag for this unit

Under Vision

Vision Signal : Does this unit have vision for the Fog of War

Radius : The radius of the Unit's vision on the Fog of War

Under Anim/Sounds

Controller : The Animation Controller for this Unit.

Idle Audio : The Audio played when the Unit is idle

Move Audio : The Audio played when the Unit is moving

Gather Audio : The Audio played when the Unit is gathering resources

Build Audio : The Audio played when the Unit is building

Attack Audio : The Audio played when the Unit is attacking

Death Object : The Object that is created when the Unit dies

A* Movement

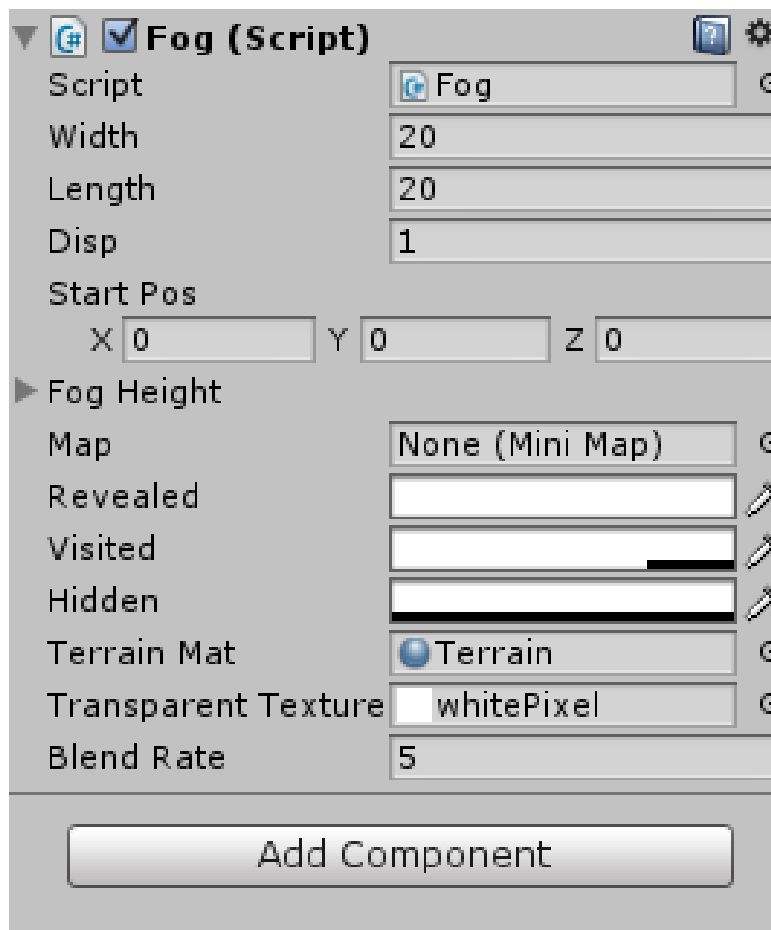
Most of the A* code is housed in the Class List script, however for it to run properly in your game you will need to add the A* Manager into the scene. Go ahead and create an Empty GameObject and apply the AStarManager component, found under Scripts/Required/AStarManager.cs. The name of the GameObject will change to A*, again just leave it.

The only option here will be the amount of threads you would like to run, which is up to you. I generally use 4 myself, just for simplicity.

Fog of War

The Fog of War system works with both normal game models and terrain objects depending on which shader you choose. For normal models, rotate them 180 degrees to get it to work properly.

To set it up, simply create a new Empty GameObject and add the Fog script found under Scripts/Fog of War/Fog.cs. The Object will be renamed to Fog when it is applied.



Variables

Width : The width of the Fog Texture

Length : The length of the Fog Texture

Disp : The displacement on the map to fill it

Start Pos : The starting position of the Fog

Revealed : The alpha layer of the revealed area. Generally set to 100% transparent

Visited : The alpha layer of the visited area. Generally set to 25% transparent

Hidden : The alpha layer of the hidden area. Generally set to 0% transparent

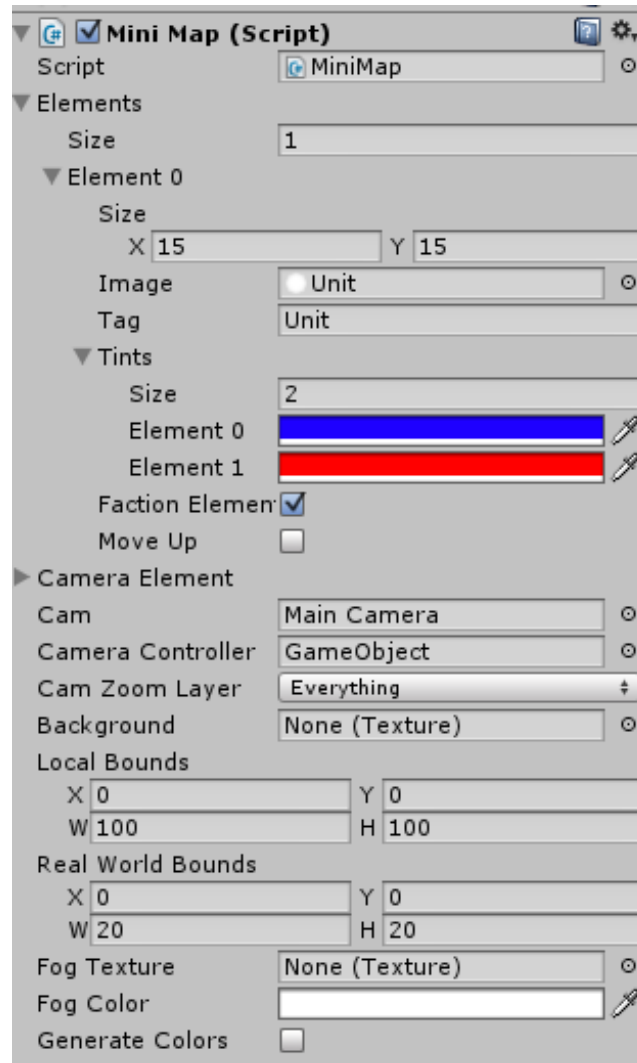
Terrain Mat : The material of the Object you want the fog to overlay

Transparent Texture : A base texture used to reset the Terrain Material when not in game. This should be preset.

Blend Rate : The rate that the Unit's vision will be updated. Play around with this, it is kinda fun.

MiniMap

The MiniMap option has quite a few features. First, you will need to create an Empty GameObject and add the MiniMap script found under Scripts/MiniMap/MiniMap.cs.



Variables

Elements : This is a custom class for the MiniMap script, with the variables Size, Image, Tag, Tints, Faction Element, and Move Up included

Size : The size of the image on the MiniMap

Image : The image to display on the MiniMap

Tag : The tag used for the Image, when setting up the unit with a MiniMapSignal script you have the option to set this as well

Tints : The tint of the Image based on Faction. Using the example above, if your Unit is under group 1, then it will display as red.

Faction Element : Should you use tints?

Move Up : This will move that element to a higher priority on the map, to display it above other elements

Camera Element : The element for the Camera visual

Cam : The Camera GameObject

Camera Controller : The Camera's parent GameObject

Camera Zoom Layer : The Layers you want the raycast to determine Camera GUI size

Background : The background of the MiniMap

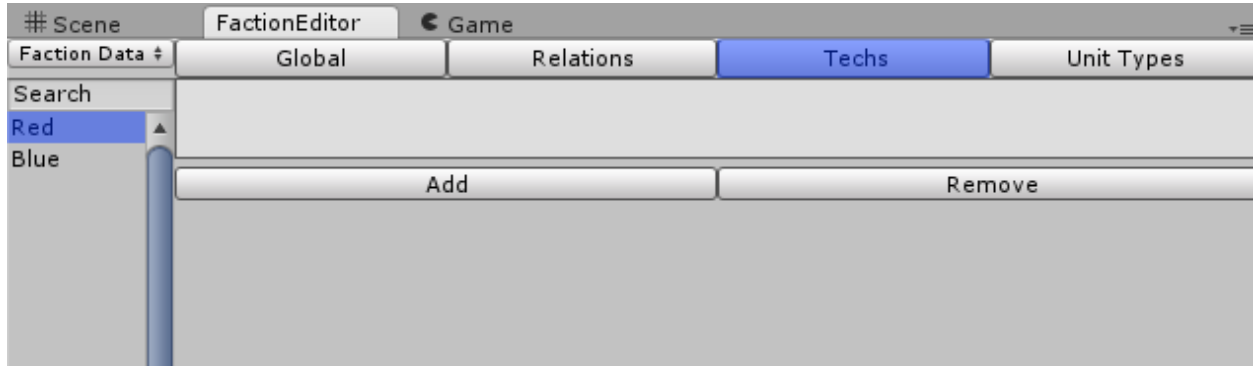
Local Bounds : The local Bounds of the MiniMap (**if you have GUI Manager set up, this will be overwritten**)

Real World Bounds : The bounds of the MiniMap in real world space

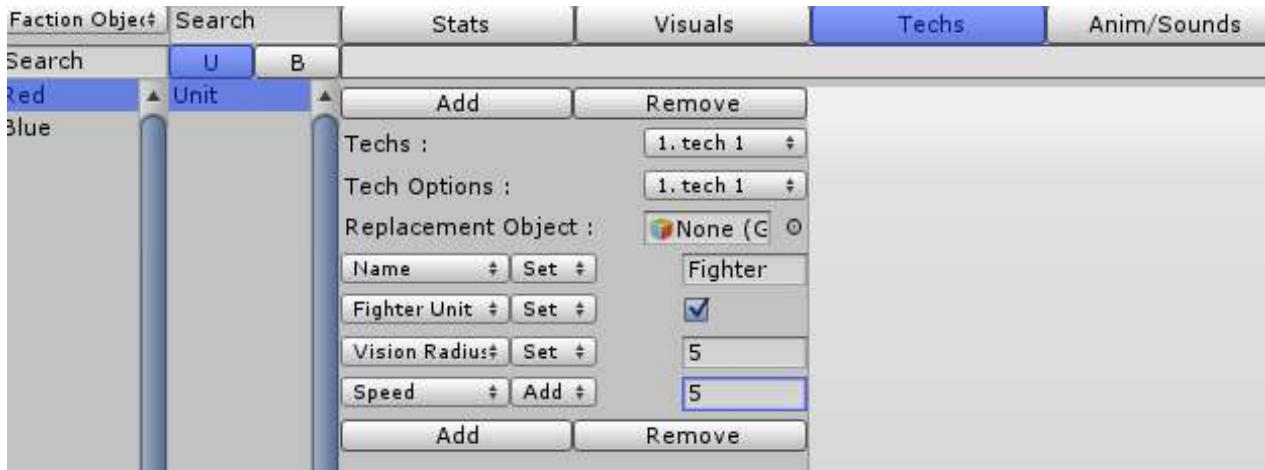
Fog Color : The tint for the Fog of War Texture overlay

Tech Tree

Alright, back to the Faction Editor to set up the tech tree. Simply go to the Faction Data option and go to Techs. Select the Faction you want to edit the techs for and begin adding techs based on what you want. You can set them to start out as active or add images to them (although adding an image to them does nothing at the moment).



Next if you go to the Unit you would like the tech to apply to, and go to Techs, then click Add, you can select the Technology you want to apply and what effects you would like this Tech to apply to this Unit type, or if you want the Unit replaced by another GameObject entirely.



Variables

Name : The name of the Technology

Image : The image of the Technology (this has no impact at the moment)

Active : Is this technology currently active?

Techs : The Unit's technology options.

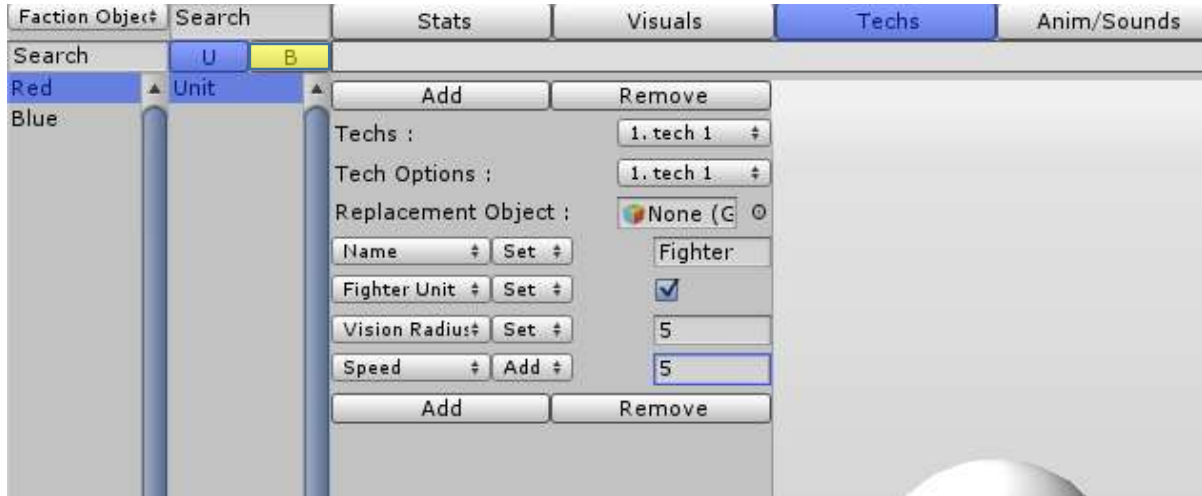
Tech Options : The Group's technology options.

Replacement Object : The object you would like to replace your GameObject with (can be left empty)

Function : Based on Getters/Setters/Adders/Subtractors, you can add any number of these and modify the Unit's variables using this

Buildings

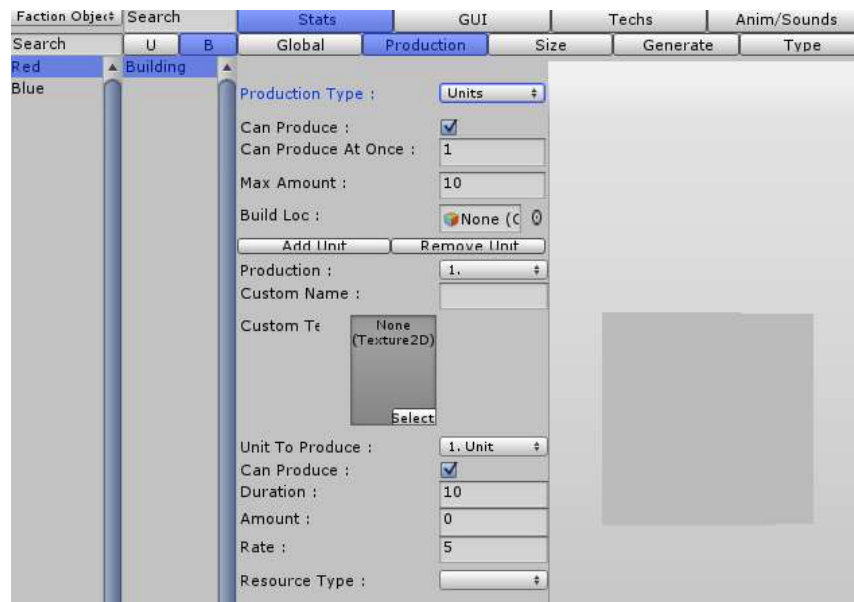
Alright, so now time to add a Building option to your game. Simply go to the Unit creation area in the Faction Editor and click the B :



Now simply add one the same way you add a Unit. Some of the options you see here are a bit different than the Unit option, however there are many familiar ones. The major differences are the Production menu, the Size menu, and the Generate menu.

Production Menu

The Production menu shows just what this building can Produce between techs and units. Simply check if you want to edit the Units or Techs production and edit the details of how it is produced there.



When a Tech has been completed, you will need to make sure that the Technology effects the Building itself so that you can then produce the next tech in the sequence.

Variables

Under Global

Auto Build : Does the Building need the in between ProgressBuilding or will it jump directly from Temporary to Final

Temporary : The display object for your Building while you are placing it, this object shouldn't have any scripts applied unless they are for visual purposes

Progress : The Building as it is being constructed, this will transition to the final building if Auto Build is false

Final : The Final Building that can be used for producing units, techs, or resources

Name : The Name of the Building

Max Health : The Maximum Health of the Building

Health : The current Health of the Building

Type : The type of the Building

Required Build Progress : The required value for the Progress Building to change to a Final Building

Starting Progress : The default starting progress when the building is placed

Progress Rate : The time it takes to add Progress per Rate to the Starting Progress

Progress per Rate : The amount to add when the Progress Rate is ready

Under Production

Quick Note : These variables (excluding build Loc) are duplicated for both Techs and Units, however you will need to change them individually

Can Produce : Can the Building Produce Techs or Units

Can Produce at Once : How many can the Building produce at once

Max Amount : How many jobs can the building retain at once

Build Loc : Where to place the Units when the building has finished them

Production : Which production option you would like to change

Custom Name : The display name of the production option

Custom Texture : The display image of the production option

Unit/Tech To Produce : The Unit or Tech to produce with this option

Can Produce : Can the building currently produce this tech?

Duration : The value it takes to produce this unit

Amount : The amount you can produce of the unit per rate

Rate : The time it takes to add the amount to the total

Resource Type : The resource option for the unit or tech to cost

Cost To Produce : The amount this unit or tech costs to produce of the specified resource

Under Size

Close Width : This is the closing width in gridpoints of the building where it is placed.

Close Length : This is the closing length in gridpoints of the building where it is placed.

Under Generate

Resource Type : The resource type to generate

Amount : The amount it will generate of that resource

Rate : The rate it will generate that amount

Under Selected

GUI Image : The image of the Building when it is selected

Objects : The objects to activate or deactivate depending on the Building's selection state

Under Production GUI

Quick Note : These variables are duplicated for Techs, Units, and Jobs however you will need to change them individually

Start Pos : The starting position of the Buttons for production

Button Size : The size of the individual buttons

Buttons Per Row : The amount of buttons per row

Displacement : The displacement on the x and y axis of the buttons

Under MiniMap

MiniMap Signal : Does this building appear on the minimap?

Tag : The MiniMap tag for this building

Group : The group for the display image, doesn't have an effect on game.

Under Vision

Vision Signal : Does this building have vision for the Fog of War

Radius : The radius of the building's vision on the Fog of War

Upward Sight Height : The distance above the unit's center that it can see through the fog

Downward Sight Height : The distance below the unit's center that it can see through the fog

Under Anim/Sounds

Controller : The Animation Controller for this building

Idle Audio : The Audio played when the building is Idle

Build Unit : The Audio played when the Building is building a unit

Build Tech : The Audio played when the Building is building a Tech

Types

Types are just Unit or Building types that you want to have other types do extra or reduced damage to. For instance, if you have a Spearman and Horseman unit, you want the Spearman unit to do more damage to the Horseman unit due to it being a counter. To create these Types, just go to the Faction Editor, Faction Data, then Types menu. Types are Universal, so you won't have to select the specific faction in order to edit these. Enter the data as you see fit, here is a simple example :

Global	Relations	Techs	Unit Types
Add Type Remove Type			
Type :	1. Type		
Type Name :	Type		
Add Strength Remove Strength			
Strength :	1. Strength Type 2		
Name :	Strength Type 2		
Target :	2. Type 2		
Ratio :	2		
Add Weakness Remove Weakness			
Weakness :	1. Weakness Type 3		
Name :	Weakness Type 3		
Target :	3. Type 3		
Ratio :	2		

To apply these types to the Unit or Building objects, go to the object you want to change in the Faction Editor and under Stats/Global you will find the Types option.

Variables

Type Name : The name of this type (ex. Swordsman, Horseman, Tower, etc)

Strength Name : The name for recognition purposes of the Strength

Target : The target of the Strength

Ratio : The multiplier for damage from the Source Type to the Target Type. (ex. If you are supposed to do 5 damage and you have a 4 multiplier you do 20 damage)

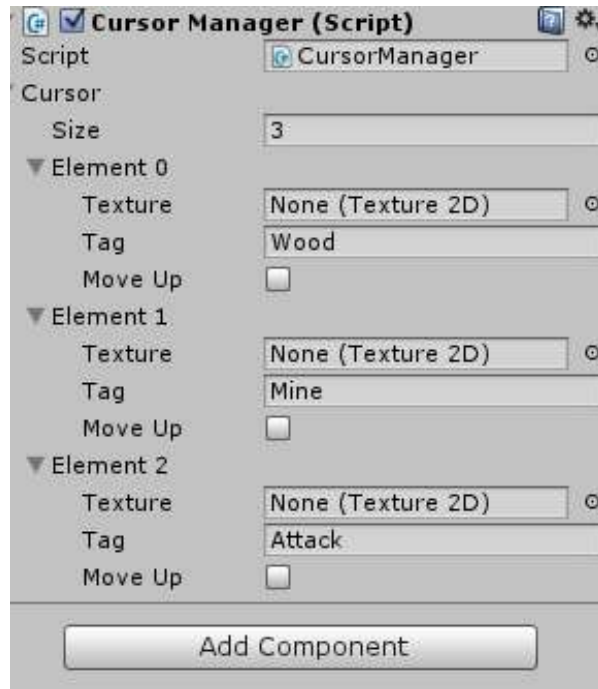
Weakness Name : The name for recognition purposes of the Weakness

Target : The target of the Weakness

Ratio : The divider for damage from the Source Type to the Target Type (ex. If you are supposed to do 10 damage and you have a 2 divider, you do 5 damage)

Cursor Manager

The Cursor Manager is fairly similar to the MiniMap for setting up. Simply set the amount of different cursor options that you want, define which you want to take precedence over which (the higher the better), define the tag you want associated and add the texture.



After that you will need to apply CursorObject to whatever objects you want to have a cursor option and set the tag to whatever cursor you want to have displayed. When the mouse hovers over that object, it will change to the cursor.

Variables

Cursor : An Array of a custom class with each element having options for Texture, Tag, and Move Up

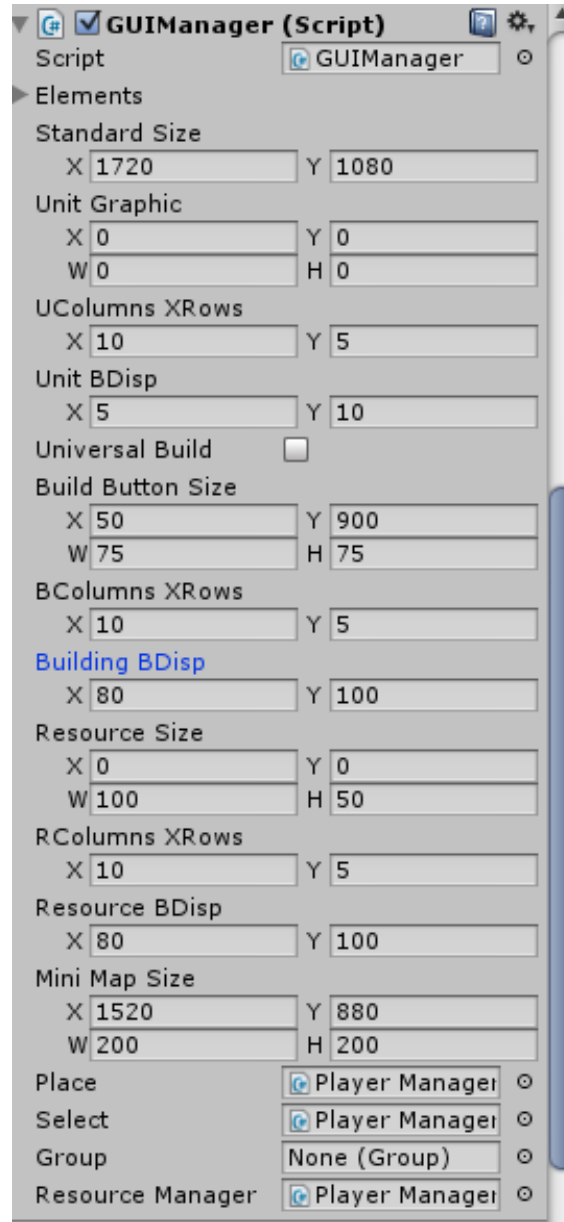
Texture : The Texture of the cursor you want to display

Tag : The Tag that the cursor is associated. When you add the Cursor Object script on to a GameObject it will react (if it has a collider) to the defined Tag

Move Up : The higher the element, the higher the priority. You can click this option to move up your option, unless it is at the top of course.

GUI Manager

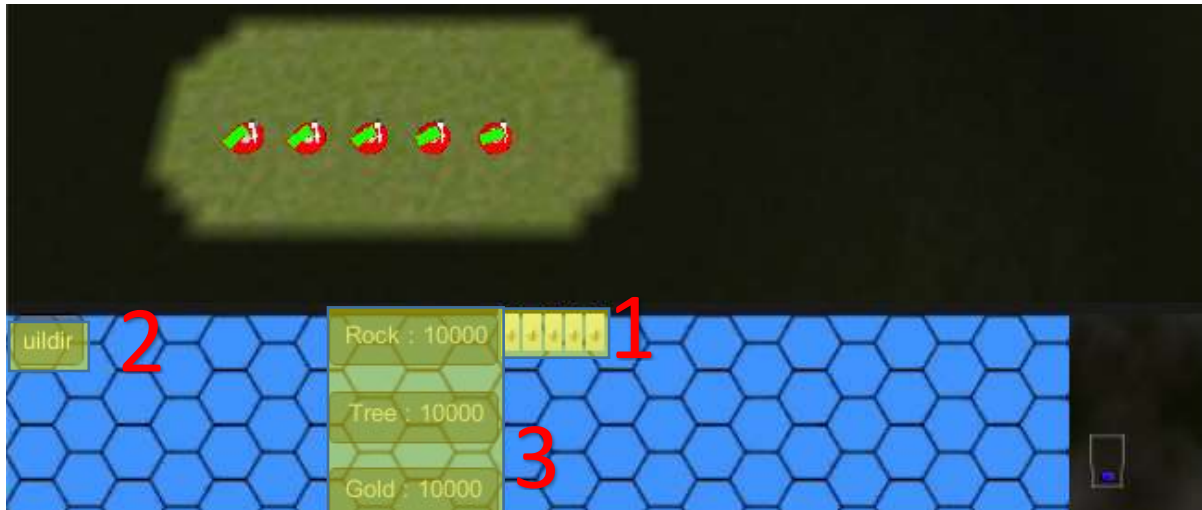
To start off, simply place the GUI Manager script on the Object with the Unit Selection script attached to it. Following that, simply edit the GUI to your specifications.



Variables

Elements : A list of GUI elements that you would like to display behind the GUI that will be scaled to the appropriate size.

Standard Size : The size you designed the GUI to fit into. If the window isn't this size, then the GUI will resize to fit the window based on the ratio.



Unit Graphic : This simply shows where you would like to display the currently selected unit's graphics, as shown by the number 1 of the above image.

UColumns XRows : The amount of columns and rows you can display.

Unit BDisp : The displacement on the x and y axis between each unit's image.

Universal Build : Determines if you can always build or if you need to select units that can build if you want to build.

Build Button Size : This shows where you display the building option, as shown in the above image at 2.

BColumns XRows : The amount of columns and rows you can display.

Building BDisp : The displacement on the x and y axis between each building's button.

Resource Size : This shows where you display the resources, as shown in the above image at 3.

RColumns XRows : The amount of columns and rows you can display.

Resource BDisp : The displacement on the x and y axis between each resource information.

MiniMap Size : This is an overload for the MiniMap's localBounds variable that will set it within the ratio size.